

Étude et développement d'un simulateur d'échanges radiatifs dans des scènes 3D statiques et dynamiques surveillées par thermographie infrarouge

Thibaud TOULLIER^{1,2}, Jean DUMOULIN^{2,1}, Laurent MEVEL^{1,2}

¹ Inria, Équipe I4S, Campus de Beaulieu, 35042 Rennes, France

² Ifsttar, COSYS-SII, Allée des Ponts et Chaussées, 44344, Bouguenais, France

Résumé - Cette étude porte sur la réalisation d'un logiciel de simulation des échanges radiatifs d'une scène modélisée en 3D. Cette simulation tient compte de différents paramètres, allant de la modélisation du capteur radiométrique ou encore l'optique de la caméra jusqu'aux conditions atmosphériques de scènes *in-situ*. La modélisation des matériaux, l'ensemble des paramètres de la simulation ainsi que la résolution des échanges radiatifs par la méthode des radiosités et sa parallélisation sur carte graphique sont présentés et discutés dans ce papier.

1. Introduction

Les récentes avancées sur les microbolomètres fonctionnant à température ambiante offrent de nouvelles perspectives pour la surveillance thermique bas coût des structures dans le génie civil. L'instrumentation de telles structures est notamment soumise à des critères économiques, des critères de sécurité ou encore des critères de robustesse. Le choix des moyens de mesures, leur positionnement et leur dimensionnement en vue d'une efficacité maximale est alors critique. De plus, les ouvrages sont aujourd'hui réalisés autour d'un BIM (*Building Information Model*) facilitant l'échange et la gestion des données du projet de construction. Dans un tel contexte, la simulation des moyens de mesure s'intégrant dans la scène observée va permettre une étude approfondie du dimensionnement des capteurs en amont de l'exploitation et ce, à moindre coût en utilisant les données de la construction. Ainsi, un simulateur proposant un bilan des échanges radiatifs et permettant d'estimer l'éclairement reçu par les détecteurs d'une caméra infrarouge a été réalisé. Une scène 3D issue par exemple d'un modèle de CAO (conception assistée par ordinateur) est chargée et visualisée directement dans le logiciel. Grâce à l'évolution des technologies de l'animation 3D ainsi que des processeurs graphiques des ordinateurs, une approche GPGPU (*general-purpose computing on graphics processing units*) permet une accélération du calcul de rendu. Cette accélération permet notamment une résolution des calculs dans des temps suffisamment raisonnables pour simuler des scènes dynamiques. L'ensemble des propriétés thermo-optiques des matériaux constituant la scène peut être défini. De même, les propriétés optiques de la caméra peuvent être spécifiées. Afin de tenir compte des conditions *in-situ*, un modèle atmosphérique [1] permettant de simuler le rayonnement solaire spectral direct et diffus reçu au sol a été intégré. De la même manière, un modèle de détecteur (CCD ou CMOS) [11] a été implémenté dans le logiciel et permet une paramétrisation avancée du modèle de la caméra. L'objectif de cette étude est de présenter la méthode de simulation d'images infrarouges multispectrales et l'ensemble des paramètres mis en jeu. Dans un premier temps, l'implémentation de l'algorithme des radiosités progressives pour établir les échanges radiatifs dans la scène est présentée. Ensuite, le modèle atmosphérique et son intégration dans le logiciel sont discutés. Le modèle de caméra infrarouge et la possibilité de paramétrer les performances des détecteurs utilisés sont à leurs tours présentés. Enfin, des exemples d'application illustrant l'utilisation du logiciel introduiront les conclusions et les perspectives de cette étude.

2. Simulation de l'éclairage d'une scène par la méthode des radiosités

2.1. Méthode des radiosités

Dans le but de simuler l'éclairage reçu au niveau des détecteurs de la caméra, le bilan radiatif de la scène est calculé par la méthode des radiosités. Cette méthode permet d'obtenir une approximation de la solution analytique de l'équation de transfert radiatif simplifiée à l'équilibre thermique grâce à la résolution d'un système linéaire. Cette méthode est qualifiée de globale car l'approximation obtenue tient compte à la fois de l'illumination directe et indirecte (réflexions successives sur l'ensemble des surfaces). Actuellement, l'implémentation de la méthode des radiosités permet de prendre en compte le rayonnement entre surfaces opaques à travers un milieu radiativement transparent. Les émissions sont supposées orthotropes et seule la diffusion (à l'exception du soleil, pour laquelle la composante spéculaire peut être considérée) est prise en compte. Enfin, le bilan radiatif est intégré sur les longueurs d'ondes considérées ; les propriétés radiatives peuvent donc être dépendantes de la longueur d'onde.

Les éléments radiatifs de la scène sont représentés par un ensemble de surfaces élémentaires. La méthode des radiosités traduit alors le bilan radiatif entre ces surfaces élémentaires sous forme d'une équation matricielle reliant les flux nets moyens échangés sur la bande spectrale considérée, les températures moyennes de ces surfaces et l'émissivité hémisphérique de ces surface.

Le facteur de forme $F_{1 \rightarrow 2}$ défini dans l'équation (1) permet de caractériser l'échange radiatif entre deux éléments diffus notés 1 et 2 [10]. Soient dA_1 et dA_2 deux éléments infinitésimaux, orientés de manière arbitraire et d'angles θ_1 et θ_2 entre les normales respectives des éléments et le segment de longueur r reliant les deux barycentres aux éléments. On appelle alors les facteurs de forme la fraction d'énergie quittant la surface dA_1 et arrivant à dA_2 . On observe que la formulation de ces facteurs de forme met en évidence la relation géométrique liant les deux surfaces (voir figure 1)

$$F_{1 \rightarrow 2} = \int_{A_1} \int_{A_2} \frac{\cos(\theta_1) \cos(\theta_2)}{\pi r^2} dA_1 dA_2 \quad (1)$$

On peut alors considérer que la radiosité B_k d'un élément k est donnée par son émission propre additionnée à la contribution de tous les autres éléments j sur la scène. Ainsi, pour une bande spectrale donnée et après discrétisation, on obtient un système d'équations linéaires :

$$B_{k,\Delta\lambda_i} = M_{k,\Delta\lambda_i} + (1 - \epsilon_{k,\Delta\lambda_i}) \sum_{j=1, j \neq k}^{j=N_{\text{elements}}} V_{kj} F_{k \rightarrow j} B_{j,\Delta\lambda_i} \quad (2)$$

où V_{kj} représente un facteur d'occlusion (présence d'autres éléments à l'intérieur de l'enceinte). Les progrès réalisés durant ces deux dernières décennies dans les processeurs graphiques, d'un point de vue logiciel comme matériel permettent aujourd'hui d'étendre la résolution par la méthode des radiosités à de grands systèmes.

2.2. Algorithme et implémentation

Afin de résoudre l'équation (2), différentes méthodes existent dans la littérature. Tout d'abord, une résolution directe par inversion de matrice est possible mais requiert beaucoup de mémoire et de temps pour de grands systèmes. En 1988, Michael F. Cohen et al. [2] ont introduit la méthode des radiosités progressives. Au lieu de résoudre entièrement le système et de devoir calculer l'ensemble des facteurs de forme pour construire la matrice à inverser, une approximation de la solution est faite en ne résolvant qu'une partie du système à chaque itération. Ainsi, seulement une partie des facteurs de forme est

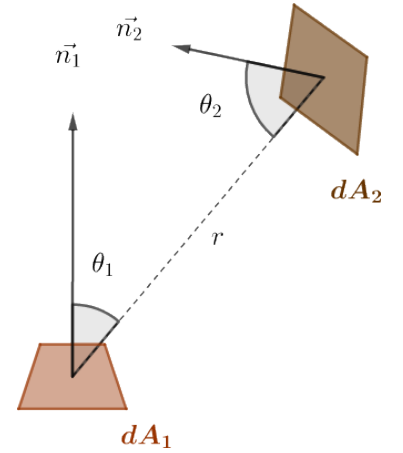


Figure 1 : Facteurs de forme pour deux éléments infinitésimaux

calculée et stockée lors d'une itération et l'algorithme peut s'arrêter lorsque l'approximation de la solution est satisfaisante, selon un critère prédéfini. Cette méthode des radiosités progressives a ainsi été implémentée dans le logiciel codé en C++. Afin de profiter de l'accélération matérielle et des dernières avancées technologiques pour le calcul, l'interface de programmation OpenGL® a été utilisée. Cette interface de programmation en C permet le calcul d'images 2D ou 3D à travers un ensemble de fonctions normalisées. Cette bibliothèque permet ainsi la manipulation de la géométrie d'objets et le rendu graphique. L'utilisation d'OpenGL® permet d'exploiter pleinement l'accélération matérielle pour le calcul des radiosités, notamment pour tenir compte de l'obstruction des objets avec un test de profondeur déjà intégré et optimisé dans la *pipeline* de rendu OpenGL®.

La résolution de l'équation (2) nécessite le calcul des facteurs de forme (1). L'utilisation d'hémicubes [3] pour cette tâche est classiquement choisie dans la littérature. Cependant, afin de réduire le nombre de rendus et éviter l'utilisation de *cubemaps* [9, 7], l'approximation des facteurs de forme est ici basée sur la méthode de la sphère unitaire intégrante de Nusselt [12]. Le facteur de forme de l'élément infinitésimal dA_1 à l'élément dA_2 est :

$$\begin{aligned} F_{d1 \rightarrow 2} &= \frac{1}{\pi} \int_{A_2} \cos(\theta_1) \frac{\cos(\theta_2) dA_2}{r^2} \\ &= \frac{1}{\pi} \int_{A_2} \cos(\theta_1) d\Omega \end{aligned} \quad (3)$$

avec $d\Omega = dA_p \frac{\cos(\theta_2) dA_2}{r^2}$. De plus, $\cos(\theta_1) dA_p$ est la projection de dA_p sur la base de l'hémisphère (voir figure 2). En intégrant alors la quantité précédente, on trouve donc :

$$F_{d1 \rightarrow 2} = \frac{1}{\pi} \int_{A_2} \cos(\theta_1) d\Omega = \frac{A_n}{\pi} \quad (4)$$

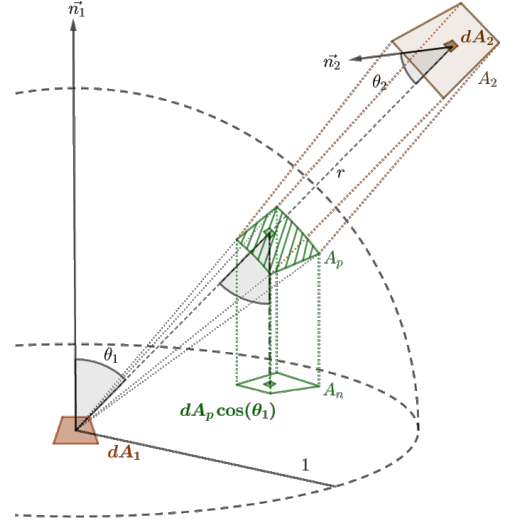


Figure 2 : Sphère intégrante de Nusselt

Fonction RADIOSITES()

Initialisation $radiosites = residus = emission$, $critereArret = \infty$, $i = elementSuivant()$

Tant que $critereArret$ non satisfait

 HemiProjection = **projectionHemisphere**(i);

Pour chaque pixels d'élément $j \in HemiProjection$

$F_{ij} = calculerFacteurForme(i, j)$;

$\Delta B = (1 - \epsilon_j) * F_{ij} * residus_i * \frac{A_i}{A_j}$;

$radiosites_j = radiosites_j + \Delta B$;

$residus_j = residus_j + \Delta B$;

$residus_i = 0$;

$i = elementSuivant()$;

//La contribution de i est réinitialisée
//choisir i tel que $B_i * A_i$ est le plus grand

Algorithme 1 : Pseudo-code pour l'algorithme des radiosités progressives

L'implémentation de l'algorithme des radiosités progressives dont le pseudo-code est présenté dans l'algorithme 1 a été réalisée par une approche GPGPU grâce aux nuanceurs de calcul (*compute shaders*) apparus dans la version 4.3 de cette interface de programmation. Avec la multiplication de l'utilisation des cartes graphiques pour le calcul scientifique ou à des fins différentes des jeux vidéos et rendus 3D, OpenGL® a offert la possibilité aux développeurs d'intervenir entièrement sur sa *pipeline*. Alors qu'auparavant les développeurs étaient cantonnés à suivre la *pipeline* de rendu et n'intervenir que sur des parties spécifiques, les nuanceurs de calcul ouvrent la voie au GPGPU permettant d'utiliser la carte graphique pour des calculs parallélisés arbitraires, qui peuvent être différents des calculs de géométrie et

de pixels. Ainsi, le logiciel utilise d'une part ces nuances de calcul pour la résolution de l'équation des radiosités et d'autres part les nuances "classiques" (*vertex/fragment shaders*) pour la génération de l'image final et l'affichage. L'ensemble des calculs sont donc fait sur le GPU minimisant ainsi les transferts de mémoire CPU-GPU (voir figure 8).

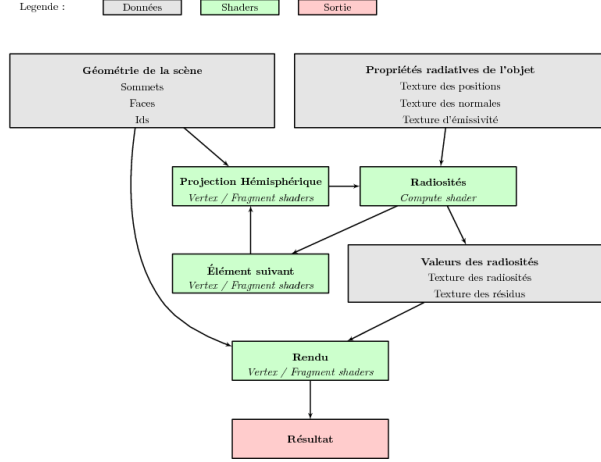


Figure 3 : Représentation des étapes principales de l'algorithme des radiosités progressives avec OpenGL®

3. Irradiance solaire spectrale

Pour modéliser les scènes *in-situ* dans le proche infrarouge en particulier, il est nécessaire de connaître l'irradiance solaire spectrale reçue pour un lieu et une heure donnés. Cette modélisation nécessite d'une part de connaître la position du Soleil dans le ciel et, d'autre part, d'utiliser un modèle atmosphérique pour tenir compte des différentes absorptions, réflexions et diffusions du rayonnement extraterrestre par l'atmosphère.

3.1. Position du soleil

La position du soleil peut être déterminée grâce à la connaissance de la longitude (L), la latitude (Φ), la date et l'heure considérées [15]. L'angle de déclinaison δ qui représente la position angulaire du Soleil au midi solaire par rapport au plan de l'équateur peut s'écrire, d'après [5] :

$$\delta = 23.45 \sin\left(\frac{284 + n}{365} \times 360\right) \quad (5)$$

D'après [15], on exprime l'heure solaire (h_{sol}) en fonction de l'heure légale qui dépend du fuseau horaire UTC+ X par :

$$h_{sol} = \frac{4}{60} \times (15 \times X - L) + \frac{E}{60} + h_{leg} \quad (6)$$

où la correction en minutes E qui provient des variations de la vitesse de la rotation de la Terre sur elle-même est donnée par [14] (voir note de page dans [15]).

L'angle de zenith est alors obtenu par la relation d'après [6] :

$$\cos(\theta_z) = \sin(\delta) \sin(\Phi) + \cos(\delta) \cos(\Phi) \cos(\omega) \quad (7)$$

Enfin l'angle d'azimut, est exprimé par la relation suivante [6] :

$$\gamma_s = \text{sign}(\omega) \left| \arccos\left(\frac{\cos(\theta_z) \sin(\Phi) - \sin(\delta)}{\sin(\theta_z) \cos(\Phi)}\right) \right| \quad (8)$$

La figure 4 montre une scène du point de vue du soleil à différents moments de l'année.

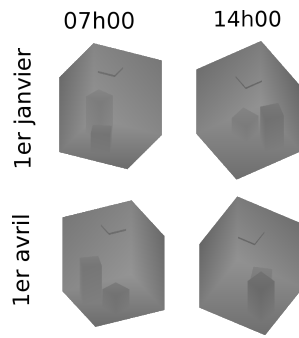


Figure 4 : Visualisation d'une scène type Cornell's Box du point de vue du soleil à deux moments de l'année différents, pour deux heures différentes de la journée. Les niveaux de gris représentent la profondeur

3.2. Irradiance spectrale solaire directe et diffuse

Le modèle d'irradiance solaire décrit dans [8] a été implémenté et permet d'estimer l'irradiance solaire spectrale reçue par la scène. La position précédemment calculée est l'un des paramètres d'entrée du modèle parmi d'autres tels que le modèle atmosphérique, le modèle d'aérosol, l'altitude, la visibilité etc. Deux exemples de données simulées par ce modèle et validant l'implémentation vis-à-vis de l'article d'origine sont montrés ci-dessous dans les figures 5 et 6 :

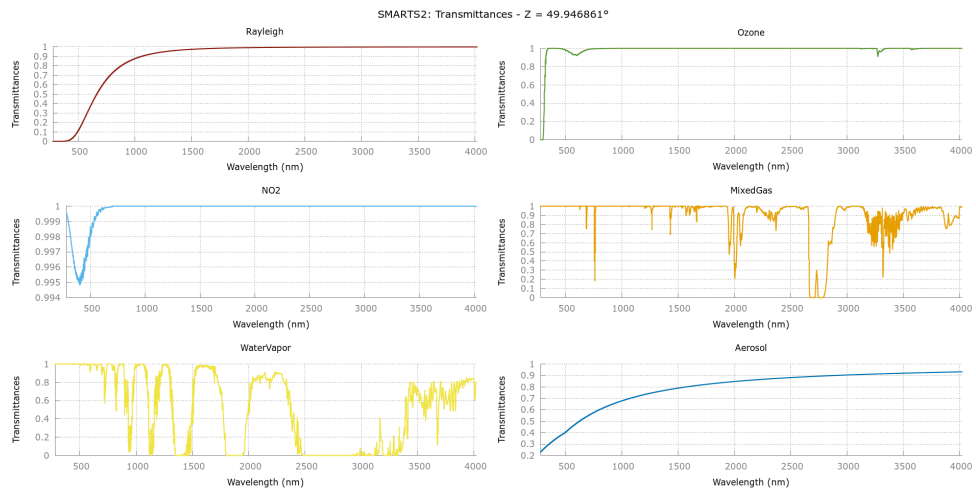


Figure 5 : Transmissions atmosphériques calculées pour un 1er janvier à 14h avec le modèle atmosphérique MLW (Mid Latitude Winter) et un modèle d'aérosols urbain

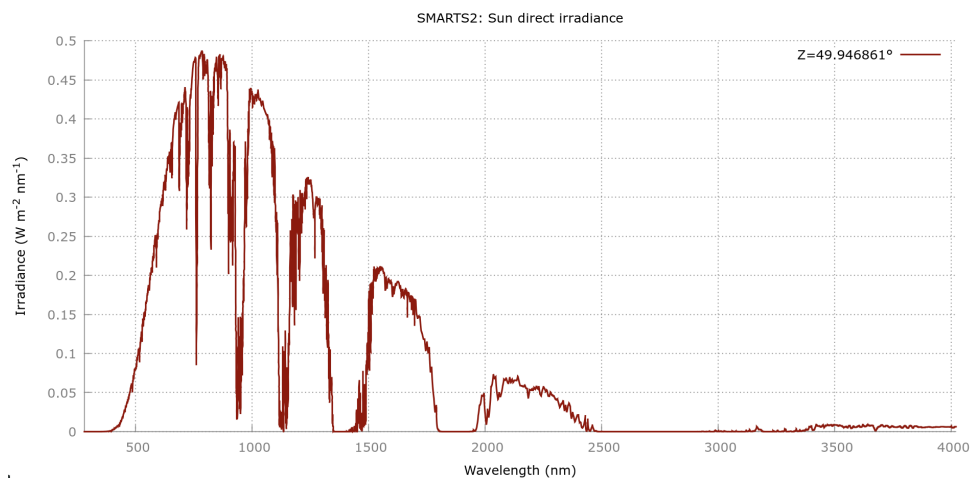


Figure 6 : Irradiance solaire spectrale directe associée reçue au sol

4. Modèle de caméra infrarouge et de détecteurs

Les paramètres de la caméra tels que sa focale, sa position, sa résolution, ou sa fréquence d'acquisition par exemple, sont des entrées du programme qui permettent d'une part de visualiser la scène par cette optique et d'autre part de calculer *in-fine* l'image infrarouge. Afin de considérer la chaîne d'acquisition de la caméra infrarouge, les modèles de capteurs CCD/CMOS définis dans [11] ont été implémentés sur GPU via des nuanciers. Le rendu de la chaîne d'acquisition (bruit de grenaille, influence des gains, convertisseur analogique-numérique etc.) est ainsi fait en temps réel, visualisable directement par l'utilisateur. La matrice qui définit les capteurs de la caméra est alors personnalisable. Chaque pixel pointe vers un capteur qui possède ses propres caractéristiques, définies dans le modèle CCD/CMOS.

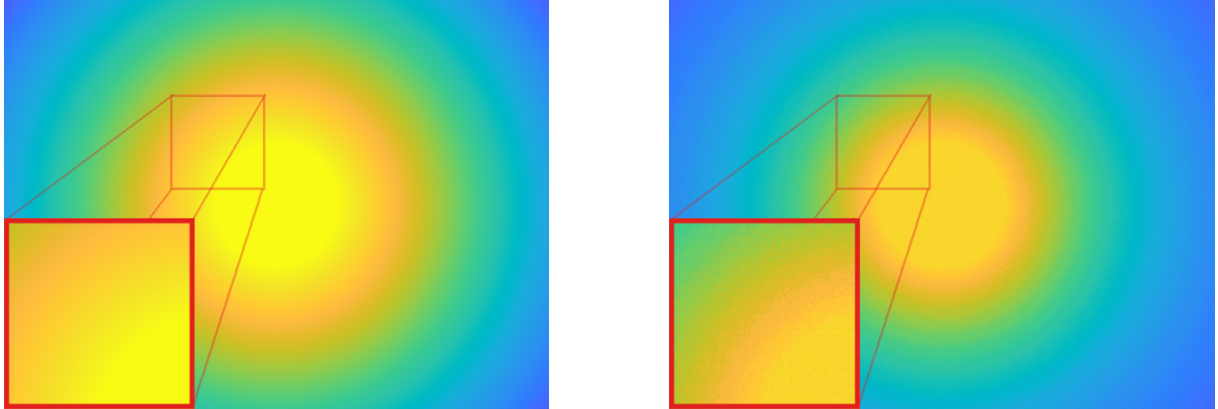


Figure 7 : Signal reçu par la caméra (gauche). Réponse à l'impulsion de la caméra, on peut observer l'effet de la numérisation du signal ainsi que le bruit dû à la chaîne d'acquisition

5. Applications

La figure 8 montre l'ensemble du processus de simulation d'images obtenues par thermographie infrarouge.

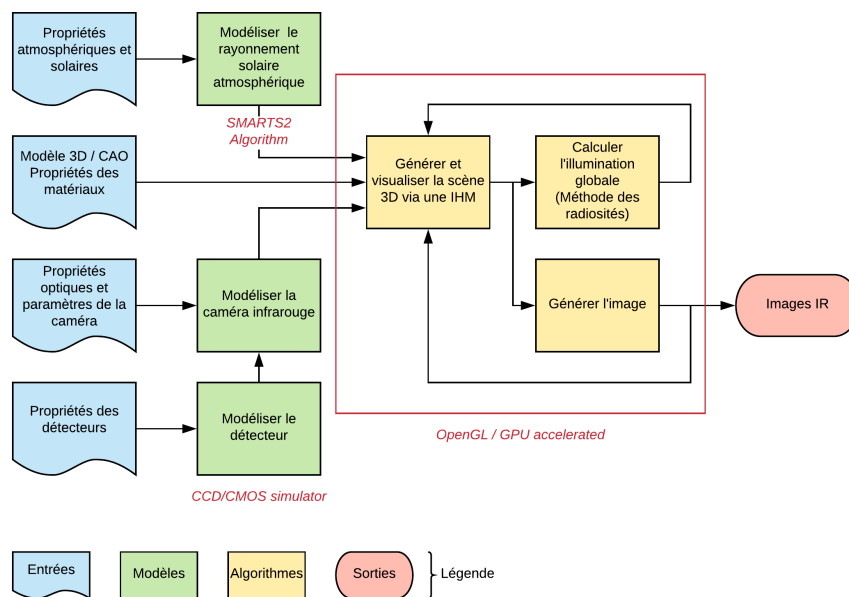


Figure 8 : Diagramme de processus du logiciel montrant l'ensemble des étapes de la simulation numérique.

La figure 9 montre un exemple de cartes des radiosités obtenues avec l'exemple d'un simple bâtiment dans le proche infrarouge. On observe la contribution du soleil et l'ombre associée à cette illumination. L'effet du maillage sur la résolution finale et la qualité de l'interpolation est tout aussi visible, et très documenté dans la littérature [4].

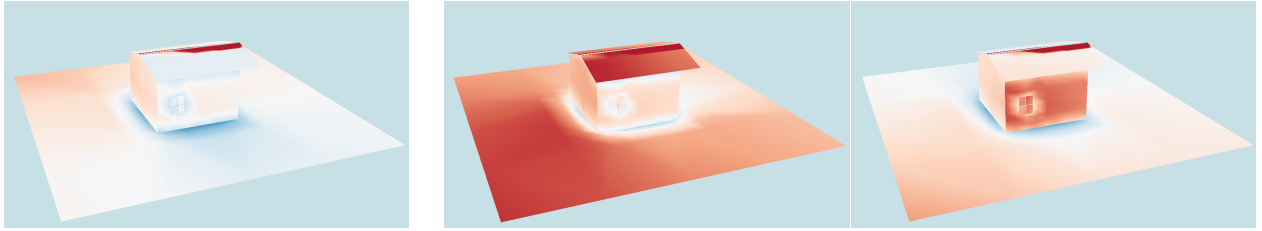


Figure 9 : Carte des radiosités pour un bâtiment la même journée, à des heures différentes (de gauche à droite : 6h, 12h et 16h)

La figure 10 suivante montre finalement la possibilité offerte par le logiciel pour simuler des scènes dynamiques. On peut ainsi observer dans les deux images suivantes, l'effet de la fréquence d'acquisition de la caméra. Dans le premier cas, la fréquence d'acquisition de la caméra est fixée à $100Hz$ pour une première vitesse de déplacement de la roue de $80km/h$, puis $130km/h$. Dans le second cas, la fréquence d'acquisition de la caméra est fixée à $1000Hz$.

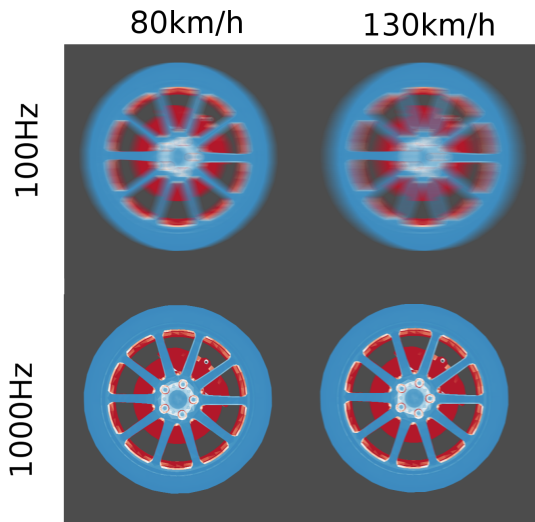


Figure 10 : Impact de la fréquence d'acquisition de la caméra pour une roue se déplaçant à différentes vitesses.

6. Conclusion et perspectives

Un programme d'imagerie infrarouge permettant de simuler les échanges radiatifs de scènes 3D statiques et dynamiques a été réalisé. L'implémentation de la méthode des radiosités sur GPU via OpenGL® permet une visualisation et un rendu parallélisé. L'utilisation de modèles de capteurs et d'irradiance spectrale solaire issues de la littérature permettent une simulation *in-situ* avancée. Les différents paramètres des modèles permettent de réaliser de multiples simulations et d'étudier l'influence de ces paramètres sur la mesure. Des améliorations dans la simulation sont cependant possibles. En effet, la méthode des radiosités implémentée, bien que parallélisée, peut être améliorée en réalisant un maillage adaptatif, se raffinant selon l'énergie reçue [7]. De même, pour pallier l'orthotropie, une émissivité bidirectionnelle pourrait être implémentée en utilisant les harmoniques sphériques par exemple [13]. Enfin, la prochaine étape sera d'améliorer le modèle optique de la caméra, en prenant en compte différentes aberrations optiques, dues à la lentille de la caméra. Un dernier objectif est de réaliser une comparaison entre la mesure par thermographie infrarouge d'une cible constituée de différents matériaux connus et contrôlée en température et la simulation associée.

Remerciements : Ce travail est partiellement financé par la région Bretagne.

Références

- [1] H. K. Askar and W. J. Batty. Assessing the performance of the “Simple Model of the Atmospheric Radiative Transfer of Sunshine” (SMARTS2) in a first tier of software using empirical weather data. *Energy*, 30(9) :1577–1588, July 2005.
- [2] Michael F. Cohen, Shenchang Eric Chen, John R. Wallace, and Donald P. Greenberg. A progressive refinement approach to fast radiosity image generation. *ACM SIGGRAPH computer graphics*, 22(4) :75–84, 1988.
- [3] Michael F. Cohen and Donald P. Greenberg. The hemi-cube : A radiosity solution for complex environments. In *ACM SIGGRAPH Computer Graphics*, volume 19, pages 31–40. ACM, 1985.
- [4] Michael F. Cohen and John R. Wallace. *Radiosity and Realistic Image Synthesis*. AP Professional, 1995.
- [5] P. I. Cooper. The absorption of radiation in solar stills. *Solar Energy*, 12(3) :333–346, January 1969.
- [6] John A Duffie and William A Beckman. *Solar Engineering of Thermal Processes*. page 928.
- [7] Greg Coombe and Mark Harris. Chapter 39. Global Illumination Using Progressive Refinement Radiosity. In *GPU Gems : Programming Techniques for High-Performance Graphics and General-Purpose Computation*, volume 2.
- [8] Christian Gueymard. Simple Model for the Atmospheric Radiative Transfer of Sunshine (SMARTS2) Algorithms and performance assessment. page 84.
- [9] Wolfgang Heidrich and Hans-Peter Seidel. *View-Independent Environment Maps*. The Eurographics Association, 1998.
- [10] J.R. Howell, R. Siegel, and M.P. Pinar. *Thermal Radiation Heat Transfer*. CRC Press, 5th edition, 2010.
- [11] Mikhail Konnik and James Welsh. High-level numerical simulations of noise in CCD and CMOS photosensors : Review and tutorial. *arXiv :1412.4031 [astro-ph]*, December 2014.
- [12] Wilhelm Nusselt. Graphische bestimmung des winkilverhältnisses bei der wärmestrahlung. *Zeitschrift des Vereines Deutscher Ingenieure*, (20) :673, 1928.
- [13] François X. Sillion and Claude Puech. *Radiosity and Global Illumination*, volume 11. Morgan Kaufmann Publishers, 1994.
- [14] J. W. Spencer. Fourier Series Representation of the Position of the Sun. *Search*, 2(5) :172, 1971.
- [15] Nicolas Le Touz. Conception et étude d’infrastructures de transports à énergie positive : de la modélisation thermomécanique à l’optimisation de tels systèmes énergétiques. page 249.